Sparse Neural Attentive Knowledge-based Models for Grade Prediction

Sara Morsy
Department of Computer Science
& Engineering
University of Minnesota
morsy@cs.umn.edu

George Karypis
Department of Computer Science
& Engineering
University of Minnesota
karypis@cs.umn.edu

ABSTRACT

Grade prediction for future courses not yet taken by students is important as it can help them and their advisers during the process of course selection as well as for designing personalized degree plans and modifying them based on their performance. One of the successful approaches for accurately predicting a student's grades in future courses is Cumulative Knowledge-based Regression Models (CKRM). CKRM learns shallow linear models that predict a student's grades as the similarity between his/her knowledge state and the target course. A student's knowledge state is built by linearly accumulating the learned provided knowledge components of the courses he/she has taken in the past, weighted by his/her grades in them. However, not all the prior courses contribute equally to the target course. In this paper, we propose a novel Neural Attentive Knowledge-based model (NAK) that learns the importance of each historical course in predicting the grade of a target course. Compared to CKRM and other competing approaches, our experiments on a large real-world dataset consisting of $\sim$1.5 grades show the effectiveness of the proposed NAK model in accurately predicting the students' grades. Moreover, the attention weights learned by the model can be helpful in better designing their degree plans.

Keywords

grade prediction, knowledge-based models, neural networks, attention networks, undergraduate education

1. INTRODUCTION

The average six-year graduation rate across four-year higher-education institutions has been around 59% over the past 15 years [9, 2], while less than half of college graduates finish within four years [2]. These statistics pose challenges in terms of workforce development, economic activity and national productivity. This has resulted in a critical need for analyzing the available data about past students in order to provide actionable insights to improve college student graduation and retention rates.

One approach for improving graduation and retention rates is to help students make more informed decisions about selecting the courses they register for in each term, such that the knowledge they have acquired in the past would prepare them to succeed in the next-term enrolled courses. Polyzou *et al.* [15] proposed course-specific linear models that learn the importance (or weight) or each previously-taken term towards accurately predicting the grade in a future course. One limitation of this approach is that in order to make accurate predictions, the model needs to have sufficient training data for each (prior, target) pair. Morsy *et al.* [13] developed Cumulative Knowledge-based Regression Models (CKRM) that also build on the idea of accumulating knowledge over time. CKRM predicts a student's grades as the similarity between his/her knowledge state and the target course. Both a student's knowledge state and a target course are represented as low-dimensional embedding vectors and the similarity between them is modeled by their inner product. A student's knowledge state is implicitly computed as a linear combination of the so-called provided knowledge component vectors of the previously-taken courses, weighted by his/her grades in them. Though CKRM was shown to provide state-of-the-art grade prediction accuracy, it is limited in that it assumes that all historical courses contribute equally in estimating the student's grade in a future course. Intuitively, students take courses from different departments, and each course would require an acquisition of knowledge from a few other courses, with different weights.

Motivated by the success of neural attentive networks in different fields [7, 12, 6, 1, 20], in this paper, we improve upon CKRM by learning the different importance of previously-taken courses in estimating the grade of a future course. We leverage the recent advances in neural attentive networks to learn these different weights, by employing both softmax and sparsemax activation functions that output posterior probabilities, i.e., attention weights, for the prior courses. The sparsemax function has an additional benefit of truncating the small probability values to zero, assigning zero effect to the irrelevant prior courses when predicting a target course's grade.

The main contributions of this work are as follows:

1. We propose a Neural Attentive Knowledge-based model

Sara Morsy and George Karypis "Sparse Neural Attentive Knowledge-based Model for Grade Prediction" In: *Proceedings of The 12th International Conference on Educational Data Mining (EDM 2019)*, Collin F. Lynch, Agathe Merceron, Michel Desmarais, & Roger Nkambou (eds.) 2019, pp. 366 - 371

(NAK) for grade prediction that improves upon CKRM by employing the attention mechanism in neural networks to learn the different importance of the prior courses towards predicting the grades of target courses. To our knowledge, this is the first work to apply attentive neural networks to grade prediction.

2. We leverage the recent sparsemax activation function for the attention mechanism that produces sparse attention weights instead of soft attention weights.

3. We performed an extensive experimental evaluation on a real world dataset obtained from a large university that spans a period of 16 years and consists of ∼1.5 grades. The results show that our proposed NAK model significantly improves the prediction accuracy compared to the competing models. In addition, the results show the effectiveness of the attention mechanism in learning the different importance of the previously-taken courses towards each target course, which can help in designing better degree plans and more informed course selection decisions.

2. DEFINITIONS AND NOTATIONS

Boldface uppercase and lowercase letters will be used to represent matrices and vectors, respectively, e.g., $\mathbf{G}$ and $\mathbf{p}$. The ith row of matrix $\mathbf{P}$ is represented as $\mathbf{p}_i^T$, and its jth column is represented as $\mathbf{p}_j$. The entry in the ith row and jth column of matrix $\mathbf{G}$ is denoted as $g_{i,j}$. A predicted value is denoted by having a hat over it (e.g., $\hat{g}$).

Matrix $\mathbf{G}$ will represent the $m \times n$ student-course grades matrix, where $g_{s,c}$ denotes the grade that student s obtained in course c, relative to his/her average previous grade. Following the row-centering technique that was first proposed by Polyzou *et al.* [15], we subtract each student's grade from his/her average previous grade, since this was shown to significantly improve the prediction accuracy of different models. As there can be some students who achieved the same grades in all their prior courses, and hence their relative grades will be zero, in this case, we assigned a small value instead, i.e., 0.01. This is to prevent a prior course from not being considered in the model computation. A student s enrolls in sets of courses in consecutive terms, numbered relative to s from 1 to the number of terms in he/she has enrolled in the dataset. A set $\mathcal{T}_{s,w}$ will denote the set of courses taken by student s in term w.

3. RELATED WORK

3.1 Grade Prediction Methods

Grade prediction approaches for courses not yet taken by students have been extensively explored in the literature [16, 17, 8, 18, 15, 13, 5]. In this section, we review some research in grade prediction that is most relevant to our work.

3.1.1 Course-Specific Regression Models (CSR)

A more recent and natural way to model the grade prediction problem is to model the way the academic degree programs are structured. Each degree program would require the student to take courses in a specific sequencing such that the knowledge acquired in previous courses are required for the student to perform well in future courses. Polyzou *et al.* [15] developed course-specific linear regression models (CSR) that build on this idea. A student's grade in a course is estimated as a linear combination of his/her grades in previously-taken courses, with different weights learned for each (prior, target) course pair. For a student s and a target course j, the predicted grade is estimated as:

$$\hat{g}_{s,j} = cb_j + \sum_{i \in \mathcal{P}} w_{i,j}\ g_{s,i}, \tag{1}$$

where cb_j is the bias terms for course j, $w_{i,j}$ is the weight of course i towards predicting the grade of course j, $g_{s,i}$ is the grade of student s in course i, and $\mathcal{P}$ is the set of courses taken by s prior to taking course j. To achieve high prediction accuracy, CSR requires sufficient training data for each (prior, target) pair, which can hinder these models from good generalization.

3.1.2 Cumulative Knowledge-based Regression Models (CKRM)

Morsy *et al.* [13] developed Cumulative Knowledge-based Regression Models (CKRM), which is also based on the fact that the student's performance in a future course is based on his/her performance in the previously-taken courses. It assumes that a space of knowledge components exists such that each course provides a subset of these components as well as requires the knowledge of some of these components from the student in order to perform well in it. The student by taking a course thus acquires its knowledge components in a way that depends on his/her grade in that course. The overall knowledge acquired by the student after taking a set of courses is then represented by a knowledge state vector that is computed as the sum of the knowledge component vectors of those courses, weighted by his/her grades in them. Let $\mathbf{p}_i$ denote the provided knowledge component vector for course i. The knowledge state vector for student s at term t can be expressed as follows:

$$\mathbf{k}_{s,t} = \sum_{w=1}^{t-1} \xi(s,w,t) \sum_{i \in \mathcal{T}_{s,w}} \Big(g_{s,i}\ \mathbf{p}_i\Big), \tag{2}$$

where $g_{s,i}$ is the grade that student s obtained on course i, and $\xi(s,w,t)$ is a time-based exponential decaying function designed to de-emphasize courses that were taken a long time ago.

Given the student's knowledge state vector prior to taking a course and that course's required knowledge component vector, denoted as r_j, CKRM estimates the student's expected grade in that course as the inner product of these two vectors, i.e.,

$$\hat{g}_{s,j} = cb_j + \mathbf{k}_{s,t}^T\ \mathbf{r}_j, \tag{3}$$

where cb_j is as defined in Eq. 1, and $\mathbf{k}_{s,t}$ is the corresponding knowledge state vector. These course-specific linear models are estimated from the historical grade data and can be considered as capturing and weighting the knowledge components that a student needs to have accumulated in order to perform well in a course.

3.2 Neural Attentive Models

Our work relies on the attention mechanism, which has been recently introduced in neural networks and was shown to improve the performance of different models and give better

explanations to the importance of different objects towards a target object [6, 20, 7, 3]. Our work leverages several advances in this area. The most commonly-used activation function for the attention mechanism is the softmax function, which is easily differentiable and gives soft posterior probabilities that normalize to 1. A major disadvantage of the softmax function is that it assumes that each object contributes to the compressed representation, which may not always hold in some domains. To solve this, we need to output sparse posterior probabilities and assign zero to the irrelevant objects. Martins *et al.* [11] proposed the sparsemax activation function, which has the benefit of assigning zero probabilities to some output variables that may not be relevant for making a decision. This is done by defining a threshold, below which small probability values are truncated to zero. We also leverage the controllable sparsemax activation function recently proposed by Laha *et al.* [10] that controls the desired degree of sparsity in the output probabilities. This is done by adding an L2 regularization term that is to be maximized in the loss function. This will potentially encourage larger probability values for some objects, moving the rest to zero.

4. PROPOSED MODEL

4.1 Motivation

Consider a sample student who is declared in a Computer Science major and is in his/her second or third year in college. Table 1 shows the set of prior courses that this student has already take and the set of courses that this student is planning on taking the next term. With CKRM (Section 3.1.2), all these prior courses would contribute equally to predicting the grade of each target course. However, we can see that, intuitively, from the courses' names, there are courses that are strongly related to each target course and other courses that are irrelevant to it. For instance, it is reasonable to expect that the Intermediate German II course is more related to the Intermediate German I course than any of the other courses that the student has already taken. Along the same lines, we expect that the Algorithms and Data Structures course is more related to other Computer Science courses, such as the Advanced Programming Principles and the Program Design and Development courses. Assuming equal contribution among these prior courses can hinder the grade prediction model from accurately learning the course representations, and hence lead to poor predictions.

4.2 Overview

In this work, we present our Neural Attentive Knowledge-based model, NAK, which predicts a students' grades in future courses by employing an attention mechanism on the prior courses. We use CKRM as the underlying model (see Section 3.1.2).

4.3 Attention-based Pooling Layer for Prior Courses

In order to learn the different contributions of the prior courses in estimating the student's grade in a future course, we can employ the CSR technique (see Section 3.1.1) that learns the importance of each prior course in estimating the grade of each future course. Thus, we would estimate a knowledge state vector for each target course j, using the following equation:

$$\mathbf{k}_{s,t,j} = \sum_{w=1}^{t-1} \sum_{i \in \mathcal{T}_w} \Big(a_{i,j} \; g_{s,i} \; \mathbf{p}_i \Big), \tag{4}$$

where $a_{i,j}$ is a learnable parameter that denotes the attention weight of course i in contributing to student s's knowledge state when predicting s's grade in course j. Note that we have removed the time-decaying function $\xi(s, w, t)$ that was used in CKRM (see Eq. 2), since it would be implicitly included in the attention weights. However, this solution requires sufficient training data for each (i, j) pair in order to be considered an accurate estimation.

In order to be able to have accurate attention weights between all pairs of prior and target courses, even the ones that do not appear together in the training data, we propose to use the attention mechanism that was recently used in neural networks [1, 19]. The main idea is to estimate the attention weight $a_{i,j}$ from the embedding vectors for courses i and j.

In order to compute the similarity between the embeddings of prior course i and target course j, we use a single-layer perceptron as follows:

$$z_{i,j} = \mathbf{h}^T \mathrm{RELU}(\mathbf{W}(\mathbf{q}_i \odot \mathbf{r}_j) + \mathbf{b}), \tag{5}$$

where $\mathbf{q}_i = g_{s,i}\mathbf{p}_i$ denotes the embedding of the prior course i, weighted by the student's grade in it, $\odot$ denotes the Hadamard product, and $\mathbf{W} \in \mathcal{R}^{l \times d}$ and $\mathbf{b} \in \mathcal{R}^l$ denote the weight matrix and bias vector that project the input into a hidden layer, respectively, and $\mathbf{h} \in \mathcal{R}^l$ is a vector that projects the hidden layer into an output attention weight, where d and l denote the number of dimensions of the embedding vectors and attention network, respectively. RELU denotes the Rectified Linear Unit activation function that is usually used in neural attentive networks.

4.3.1 Softmax Activation Function

The most common activation function used for computing these attention weights is the softmax function [19]. Given a vector of real weights $\mathbf{z}$, the softmax activation function converts it to a probability distribution, which is computed component-wise as follows:

$$\mathrm{softmax}_i(\mathbf{z}) = \frac{\exp(z_i)}{\sum_j \exp(z_j)}. \tag{6}$$

We will refer to the NAK model that uses the softmax activation function as **NAK(soft)**.

4.3.2 Sparsemax Activation Function

Although the softmax activation function has been used to design attention mechanisms in many domains [14, 1, 6, 12, 7], we believe that using it for grade prediction is not optimal. Since a student enrolls in several courses, and each course requires knowledge from one or a few other courses, we hypothesize that some of the prior courses should have no effect, i.e., zero attention, towards predicting a target course's grade. We thus leverage a recent advance, the sparsemax activation function [11], to learn sparse attention weights. The idea is to define a threshold, below

Table 1: Sample of prior and target courses for a Computer Science student at University X.

Prior Courses	Target Course
Calculus I, Beginning German, Operating Systems, Intermediate German I, University Writing, Introductory Physics, Peotics in Film, Program Design & Development, Philosophy, Linear Algebra, Internet Programming, Stone Tools to Steam Engines, Advanced Programming Principles, Computer Networks	Intermediate German II
	Probability & Statistics
	Algorithms & Data Structures

which small probability values are truncated to zero. Let $\triangle^{K-1} := \{\mathbf{x} \in \mathbb{R}^K | \mathbf{1}^T\mathbf{x} = 1, \mathbf{x} \geq \mathbf{0}\}$ be the $(K-1)$-dimensional simplex. The sparsemax activation function tries to solve the following equation:

$$\text{sparsemax}(\mathbf{z}) = \underset{\mathbf{x}\in\triangle^{K-1}}{\text{argmin}} \ \|\mathbf{x} - \mathbf{z}\|^2, \tag{7}$$

which, in other words, returns the Euclidean projection of the input vector $\mathbf{z}$ onto the probability simplex.

In order to obtain different degrees of sparsity in the attention weights, Laha *et al.* [10] developed a generic probability mapping function for the sparsemax activation function, which they called **sparsegen**, and is computed as follows:

$$\text{sparsegen}(\mathbf{z}; \gamma) = \text{argmin} \ \|\mathbf{x} - \mathbf{z}\|^2 - \gamma\|\mathbf{x}\|^2, \tag{8}$$

where $\gamma < 1$ controls the L2 regularization strength of $\mathbf{x}$. An equivalent formulation for sparsegen was formed as:

$$\text{sparsegen}(\mathbf{z}; \gamma) = \text{sparsemax}\Big(\frac{\mathbf{z}}{1-\gamma}\Big), \tag{9}$$

which, in other words, applies a temperature parameter to the original sparsemax function. Varying this temperature parameter can change the degree of sparsity in the output variables. By setting $\gamma = 0$, sparsegen becomes equivalent to sparsemax. We will refer to the NAK model that uses the sparsegen activation function as **NAK(sparse)**.

4.4 Prediction

NAK then predicts the grade for student s in course j that he/she takes at term t as:

$$\hat{g}_{s,j} = cb_j + \mathbf{k}_{s,t,j}^T \ \mathbf{r}_j. \tag{10}$$

4.5 Optimization

We use the mean squared error (MSE) loss function to estimate the parameters of NAK. We minimize the following regularized RMSE loss:

$$L = -\frac{1}{2N} \sum_{s,c\in\mathbf{G}} (g_{s,c} - \hat{g}_{s,c})^2 + \alpha||\Theta||^2, \tag{11}$$

where N is the number of grades in $\mathbf{G}$. The hyper-parameter α controls the strength of L2 regularization to prevent overfitting, and $\Theta = \{\{\mathbf{cb}\}, \{\mathbf{p}_i\}, \{\mathbf{r}_i\}, \mathbf{W}, \mathbf{b}, \mathbf{h}\}$ denotes all trainable parameters of NAK.

The optimization problem is solved using AdaGrad algorithm [4], which applies an adaptive learning rate for each parameter. It randomly draws mini-batches of a given size from the training data and updates the related model parameters. The source code can be found here: `https://urlzs.com/iH8G`.

5. EVALUATION METHODOLOGY

5.1 Dataset

The data used in our experiments was obtained from the University of Minnesota (UMN), which includes 96 majors from 10 different colleges, and spans the years 2002 to 2017. At the University, the letter grading system used is A–F, which is converted to the 4–0 scale using the standard letter grade to GPA conversion. We removed any grades that were taken as pass/fail. The final dataset includes $\sim 54,000$ students, $5,800$ courses, and $1,450,000$ grades in total.

5.2 Generating Training, Validation and Test Sets

At UMN, there are three terms, Fall, Summer and Spring. We used the data from 2002 to Spring 2015 (inclusive) as the training set, the data from Spring 2016 to Fall 2016 (inclusive) as the validation set, and the data from Summer 2016 to Summer 2017 (inclusive) as the test set. For a target course taken by a student to be predicted, that student must have taken at least four courses prior to the target course, in order to have sufficient data to compute the student's knowledge state vector. We excluded any courses that do not appear in the training set from the validation and test sets.

5.3 Comparison Methods

We compared the performance of our NAK model against the following grade prediction approaches:

1. **Matrix Factorization (MF):** This approach predicts the grade for student s in course i as:
$$\hat{g}_{s,i} = \mu + sb_s + cb_i + \mathbf{u}_s^T \ \mathbf{v}_i, \tag{12}$$
where μ, sb_s and cb_i are the global, student and course bias terms, respectively, and $\mathbf{u}_s$ and $\mathbf{v}_i$ are the student and course latent vectors, respectively. We used the squared loss function with L2 regularization to estimate this model.

2. **KRM(sum):** This is CKRM the method described in Section 3.1.2.

3. **KRM(avg):** This is similar to the KRM(sum) method, except that the prior courses' embeddings are aggregated with mean pooling instead of summation. It was shown in later studies, e.g. [17], that it performs better than KRM(sum).

We implemented KRM(sum) and KRM(avg) with a neural network architecture and optimization similar to that of NAK.

Table 2: **Comparison between the baseline and proposed models.**

Model	Parameters					RMSE	PTA0	PTA1	PTA2
MF	16	1E-04	1E-02	–	–	0.724	25.7	58.6	79.5
KRM(sum)	32	1E-07	7E-04	0.3	–	0.584	32.6	70.1	87.7
KRM(avg)	32	1E-07	7E-04	0.0	–	0.584	34.9	70.6	87.7
NAK(soft)	32	1E-07	7E-04	3	–	0.589 (-0.9%)	<u>35.3</u>† (1.1%)	71.8 (1.7%)	88.0† (0.3%)
NAK(sparse)	32	1E-07	7E-04	4	0.5	<u>0.574</u>†‡ (1.7%)	<u>35.3</u>† (1.1%)	<u>72.1</u> (2.1%)	<u>88.7</u>† (1.1%)

The Parameters columns denote the following model parameters that were selected: for MF, the parameters are: the number of latent dimensions, the L2 regularization parameter, and the learning rate; for KRM(sum) and KRM(avg), the parameters are: the embedding size for courses, the L2 regularization parameter, the learning rate, and the time-decaying parameter λ; for NAK, the parameters are: the embedding size for courses, the L2 regularization parameter, the learning rate, and the number of latent dimensions for the MLP attention mechanism; and for NAK(sparse), the last parameter denotes the L2 regularization parameter γ for the sparsegen activation function. Underlined entries represent the best performance in each metric. The † and ‡ symbols are used to denote results that are statistically significant over the best performing baseline metric, and NAK(soft), respectively, using the Student's paired t-test with a p-level < 0.1. Numbers in parentheses denote the percentage of improvement over the best baseline value in each metric.

5.4 Model Selection

We performed an extensive search on the parameters of the proposed and baseline models to find the set of parameters that gives us the best performance for each model.

For all proposed and competing models, the following parameters were used. The number of latent dimensions for course embeddings was chosen from the set of values: {8, 16, 32}. The L2 regularization parameter was chosen from the values: {1e-5, 1e-7, 1e-3}. Finally, the learning rate was chosen from the values: {0.0007, 0.001, 0.003, 0.005, 0.007}. For the proposed NAK models, the number of latent dimensions for the MLP attention mechanism was selected in the range [1, 4]. For KRM(sum) and KRM(avg), the time-decaying parameter λ was chosen from the set of values: {0, 0.3, 0.5, 0.7, 1.0}.

The training set was used for estimating the models, whereas the validation set was used to select the best performing parameters in terms of the overall MSE of the validation set.

5.5 Evaluation Methodology and Metrics

The grading system used by the University uses a 12 letter grade system (i.e., A, A-, B+, ... F). We will refer to the difference between two successive letter grades (e.g., B+ vs B) as a *tick*. We converted the predicted grades into their closest letter grades. We assessed the performance of the different approaches based on the Root Mean Squared Error (RMSE) as well as how many ticks away the predicted grade is from the actual grade, which is referred to as *Percentage of Tick Accuracy*, or PTA. We computed the percentage of grades predicted with no error (zero tick), within one tick, and within two ticks, which will be referred to as PTA0, PTA1, and PTA2, respectively.

6. EXPERIMENTAL RESULTS

6.1 Performance of the Proposed Models

Table 2 shows the performance of our proposed models. Using the sparsegen activation function instead of the softmax activation function improves the prediction accuracy, with a statistically significant improvement. This shows that using the sparsegen activation function to output sparse attention weights for the prior courses achieves better prediction accuracy than producing soft probabilities for all of them. This is expected, since the student's prior courses may not be all relevant to the target course, as illustrated in Table 1.

6.2 Performance against Competing Methods

Table 2 also shows the performance of the competing models. Among the baseline methods, both KRM(sum) and KRM(avg) outperform MF. KRM(avg) outperforms KRM(sum) in PTA0 and PTA1. Both NAK(soft) and NAK(sparse) outperform all baseline methods. Even though the RMSE results of NAK(soft) is worse than these of the KRM variants, it achieved ∼1%, ∼2% and 0.5% more accurate predictions within no, one, and two tick errors, respectively. Among all baseline and proposed methods, our NAK(sparse) model outperforms all baseline methods significantly, with achieving ∼2% lower RMSE, and ∼1% more accurate predictions within two ticks than KRM(avg). This shows that using the attention-based pooling layer on the prior courses to accumulate them can better predict the grades of students in their future courses.

6.3 Qualitative Analysis on the Prior Courses Attention Weights

Recall the motivational example for the Computer Science student, discussed in Section 4.1. This student had a set of prior courses and three target courses that we would like to predict his/her grades in (See Table 1). Using KRM(sum) or KRM(avg), all the prior courses would contribute equally to the prediction of each target course. Using our proposed NAK(sparse) model, the attention weights for the prior courses with each target course are shown in Table 3[1].

We can see that, using the sparsegen activation function, only a few prior courses are selected with non-zero attention weights, which are the most relevant to each target course.

For the Intermediate German II course, we can see that the student's grade in it is most affected by two courses: the Intermediate German I course, and the University Writing course. The Intermediate German I course is listed as a pre-requisite course for the Intermediate German II course. Though the University Writing course is not listed as a pre-requisite course, after further analysis, we found out that the Intermediate German II course requires process-writing essays and are considered part of the grading system. Though the German courses are not part of the student's degree program, and are taken by a small percentage of Computer

[1]These results were obtained by learning NAK models to estimate the actual grades and not the row-centered grades. Also, we used $\mathbf{q}_i = \mathbf{p}_i$ in Eq. 5. This allowed us to get more interpretable results.

Table 3: The attention weights of the prior courses with each target course learned by NAK(sparse) for the sample student from Table 1.

Prior Courses	Target Course
Intermediate German I: 0.6980, University Writing: 0.3020	Intermediate German II
Calculus I: 0.4737, Physics: 0.3794, Program Design & Development: 0.0717, Operating Systems: 0.0497, Computer Networks: 0.0255	Probability & Statistics
Operating Systems: 0.2927, Advanced Programming Principles: 0.2582, Linear Algebra: 0.2313, Physics: 0.2178	Algorithms & Data Structures

Prior courses are sorted in non-increasing order w.r.t. to their attention weights with each target courses for clarity purposes.

Science students, our NAK model was able to learn accurate attention weights for them.

The other two target courses, Probability and Statistics, and Algorithms and Data Structures, have totally different prior courses with the largest attention weights, which are more related to them.

These results illustrate that our proposed NAK model was able to uncover the listed as well as the hidden/informal pre-requisite courses without any supervision given to the model.

7. CONCLUSION

In this work, we presented a method to improve the grade prediction accuracy, by learning the weights of the prior courses towards predicting the grade of each target course. To this end, we employed the attention mechanism on the prior courses that learns the different contributions of these courses towards each target course. We employed both a softmax and a sparsemax activation function that produce soft and sparse attention weights, respectively. The proposed models are able to capture the listed as well as the hidden pre-requisite courses for the target courses, which can be better used to design better degree plans. Our experiments showed that our models significantly outperformed the competing methods, indicating the value of the attention mechanism on the prior courses.

Acknowledgement

This work was supported in part by NSF (1447788, 1704074, 1757916, 1834251), Army Research Office (W911NF1810344), Intel Corp, and the Digital Technology Center at the University of Minnesota. Access to research and computing facilities was provided by the Digital Technology Center and the Minnesota Supercomputing Institute, `http://www.msi.umn.edu`.

8. REFERENCES

[1] D. Bahdanau and *et al.*. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[2] J. Braxton and *et al.*. *Understanding and Reducing College Student Departure: ASHE-ERIC Higher Education Report, Volume 30, Number 3*, volume 16. John Wiley & Sons, 2011.

[3] J. Chen and *et al.*. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *SIGIR*. ACM, 2017.

[4] J. Duchi and *et al.*. Adaptive subgradient methods for online learning and stochastic optimization. *JMLR*, 12(Jul):2121–2159, 2011.

[5] A. Elbadrawy and *et al.*. Domain-aware grade prediction and top-n course recommendation. In *RecSys*. ACM, 2016.

[6] X. He and *et al.*. Neural factorization machines for sparse predictive analytics. In *SIGIR*. ACM, 2017.

[7] X. He and *et al.*. Nais: Neural attentive item similarity model for recommendation. *TKDE*, 30(12):2354–2366, 2018.

[8] Q. Hu and *et al.*. Course-specific markovian models for grade prediction. In *PAKDD*. Springer, 2018.

[9] G. Kena and *et al.*. The condition of education 2016. nces 2016-144. *National Center for Education Statistics*, 2016.

[10] A. e. Laha. On controllable sparse alternatives to softmax. In *NIPS*, pages 6423–6433, 2018.

[11] A. e. Martins. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *ICML*, pages 1614–1623, 2016.

[12] L. Mei and *et al.*. An attentive interaction network for context-aware recommendations. In *CIKM*. ACM, 2018.

[13] S. Morsy and *et al.*. Cumulative knowledge-based regression models for next-term grade prediction. In *SDM*. SIAM, 2017.

[14] A. Parikh and *et al.*. A decomposable attention model for natural language inference. In *EMNLP*, 2016.

[15] A. Polyzou and *et al.*. Grade prediction with course and student specific models. In *PAKDD*. Springer, 2016.

[16] Z. Ren and *et al.*. Grade prediction with temporal course-wise influence. In *EDM*, 2017.

[17] Z. Ren and *et al.*. Ale: Additive latent effect models for grade prediction. In *SDM*. SIAM, 2018.

[18] M. Sweeney and *et al.*. Next-term student performance prediction: A recommender systems approach. *EDM*, 8(1):22–51, 2016.

[19] A. e. Vaswani. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[20] J. Xiao and *et al.*. Attentional factorization machines: learning the weight of feature interactions via attention networks. In *IJCAI*. AAAI Press, 2017.